Using the SUP without the SUP Client

Category: File Transfers

DRAFT

This article is being reviewed for completeness and technical accuracy.

Introduction

The SUP client is the recommended approach to using the SUP. The client requires Perl, however, thus may not be suitable for all purposes. The only software actually required to use the SUP is SSH. This page details the manual steps required to use the SUP with only SSH. Users should still review the client instructions for a full overview of the SUP.

SUP Manual Usage Summary

The steps below demonstrate how to get up and running with the SUP without the client using a bbftp transfer to cfe1 as an example. Consult the link in each step for full details (or simply read this page to completion).

1. Initialize a long-term key on sup-key.nas.nasa.gov (one time)

```
ssh -x -oPubkeyAuthentication=no sup-key.nas.nasa.gov \
    mesh-keygen --init <~/.ssh/authorized_keys
```

2. Generate a SUP key (one time per week)

```
eval `ssh-agent`
ssh-add ~/.ssh/id_rsa
ssh -A -oPubkeyAuthentication=no sup.nas.nasa.gov \
    mesh-keygen |tee ~/.ssh/supkey`
ssh-agent -k
```

3. Authorize host for SUP operations (one time per host)

```
ssh cfe1
touch ~/.meshrc
```

4. Authorize directories for writes (one or more times per host)

```
ssh cfe1
echo /tmp >>~/.meshrc
```

5. Prepare the SUP key for use (one time per session)

```
eval `ssh-agent`
ssh-add -t 1w ~/.ssh/supkey
```

6. Execute command (each time)

```
bbftp -L "ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q" \
    -e "put /foo/bar /tmp/c_foobar" cfe1.nas.nasa.gov
```

7. Troubleshoot problems (as needed)

SUP Key Generation

1. **On the very first use only**, invoke the "mesh-keygen" command with the "--init" option on sup-key.nas.nasa.gov to upload an SSH authorized_keys file (used *only* during key generation and revocation). An authorized_keys file contains one or more SSH public keys that allow the corresponding SSH private keys to be used for authentication to a system. The uploaded authorized_keys file can be an existing file (such as your ~/.ssh/authorized_keys file from any host) or one created specifically for this purpose using a new SSH key pair generated with ssh-keygen. The public keys in this file must be in OpenSSH format (i.e. *not* the format of the commercial SSH version used on the Secure Front-Ends [SFEs]) and must not contain any forced commands (i.e. "command="). For example, to upload an existing authorized_keys file, the following can be invoked:

   ```
   ssh -x -oPubkeyAuthentication=no sup-key.nas.nasa.gov \
       mesh-keygen --init <~/.ssh/authorized_keys
   ```

   You will be prompted to authenticate using both a password (originally your Lou password) and securID passcode (PIN + tokencode).

   Users who have never connected to sup-key.nas.nasa.gov before may need to add a "-oStrictHostKeyChecking=ask" option to the scp command line. (RSA key fingerprint of sup-key.nas.nasa.gov is 1b:9a:82:2b:b9:b0:7d:e5:08:50:1d:e8:14:76:a2:2e)

   Note that this is on sup-key *only* and that you must use the "-oPubkeyAuthentication=no" option as shown. Users outside NAS may need to add an appropriate SSH option to set their login name, such as "-l username".
2. Start an SSH agent (or use one currently running):

   ```
   eval `ssh-agent -s` (if your shell is sh/bash)
   ```

   or

   ```
   eval `ssh-agent -c` (if your shell is csh/tcsh)
   ```

3. Add a private key corresponding to one of the public keys in the authorized_keys file of step 1 to the agent (this is unnecessary if an agent is already running with the key loaded). For example:

   ```
   ssh-add ~/.ssh/id_rsa
   ```

4. Invoke the "mesh-keygen" command on sup.nas.nasa.gov. You will be prompted to authenticate using both password (originally your Lou password) and securID passcode (PIN + tokencode). After successful authentication, the mesh-keygen command prints a SUP key to your terminal, which should be saved to a file in a directory that is readable only by you. This key can be saved to a file by cut-and-paste, redirecting standard output, or using the "tee" command. For example, to generate a key and redirect it into a file starting with ~/.ssh/supkey and labeled with the current time, the following can be invoked:

   ```
   ssh -A -oPubkeyAuthentication=no sup.nas.nasa.gov \
       mesh-keygen |tee ~/.ssh/supkey.`date +%Y%m%d.%H%M`
   ```

   Users who have never connected to sup.nas.nasa.gov before may need to add a "-oStrictHostKeyChecking=ask" option to the SSH command line. (RSA key fingerprint of sup.nas.nasa.gov is 52:f3:61:9b:9c:73:79:4d:22:cb:f3:cd:9a:29:4e:fe)

   Note that you must use the "-oPubkeyAuthentication=no" option as shown. Users outside NAS may need to add an appropriate SSH option to set their login name, such as "-l username".
5. Protect your keys. In order to perform unattended operations, SUP keys cannot be encrypted, thus should always be protected with appropriate file system permissions (i.e. 400 or 600). Check the permissions of your key immediately after generation and modify if necessary. You are responsible for the privacy of your keys.

SUP Key Management

Each invocation of mesh-keygen creates a new SUP key that is valid for one week from the time of generation. Users may have multiple keys at once that all expire at different times. To facilitate the management of multiple SUP keys, the "mesh-keytime" and "mesh-keykill" commands are available.

Mesh-keytime

To determine the expiration time of a SUP key stored in a file "/key/file", the following can be invoked:

```
ssh -xi /key/file -oIdentitiesOnly=yes -oBatchMode=yes \
    sup.nas.nasa.gov mesh-keytime
```

The key fingerprint and expiration time will be printed to your terminal.

Mesh-keykill

To invalidate a specific SUP key stored in a file "/key/file" before its expiration time has passed, you must have an SSH agent running with the same key you use to generate SUP keys as described in steps 2 and 3 of the SUP Key Generation section. After which, the following can be invoked:

```
ssh -Axi /key/file -oIdentitiesOnly=yes -oBatchMode=yes \
            sup.nas.nasa.gov mesh-keykill
```

To invalidate all currently valid SUP keys, the following can be invoked:

```
ssh -Ax -oPubkeyAuthentication=no sup.nas.nasa.gov mesh-keykill --all
```

In this case, you will be prompted to authenticate using both password and securID passcode.

SUP Key Preparation

Currently, the only operations allowed with a SUP key are scp, sftp, bbftp, qstat, rsync, and test. For all operations, an SSH agent must be started with the SUP key loaded, which can be scripted as needed, because the key is unencrypted.

1. Start an SSH agent:

   ```
   eval `ssh-agent -s` (if your shell is sh/bash)
   ```

   or

   ```
   eval `ssh-agent -c` (if your shell is csh/tcsh)
   ```

2. Add a SUP key to the agent (this is the *only* key required to perform unattended SUP operations):

   ```
   ssh-add /key/file
   ```

   Since SUP keys have a lifetime of one week, the "-t" option may be used to automatically remove the key from the agent after a week has elapsed:

   ```
   ssh-add -t 1w /key/file
   ```

   The will prevent a buildup of keys in the agent, which can cause login failure as described in the SUP Troubleshooting section. Keys may be explicitly removed from the agent using the following:

   ```
   ssh-keygen -y -f /key/file >/key/file.pub
   ssh-add -d /key/file
   ```

3. Make sure agent forwarding and batch mode are enabled in your SSH client. The examples below include the appropriate options to enable agent forwarding ("-A") and batch mode ("-oBatchMode=yes").

SUP Commands

Examples of the use of each command that may be executed through the SUP are given below. Note that SUP commands must be authorized for execution on each target host and transfers to a given host must be authorized for writes.

- **bbftp** (man page)

```
bbftp -L "ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q" \
    -e "put /foo/bar /tmp/c_foobar" cfe1.nas.nasa.gov
```

  Note that **you must use the fully-qualified domain name of the target host** (in this case, cfe1.nas.nasa.gov) if you are not within the NAS domain.
- **bbscp** (man page)

```
bbscp -L "ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q" \
    foobar cfe1.nas.nasa.gov:/tmp/c_foobar
```

  Note that bbscp is just a client-side wrapper for bbftp, thus like bbftp, **you must use the fully-qualified domain name of the target host** (in this case, cfe1.nas.nasa.gov) if you are not within the NAS domain.
- **qstat** (man page available on Pleiades and Columbia))

```
ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q cfe1 qstat @pbs1
```

- **rsync** (man page)

```
rsync -e "ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q" \
    foobar cfe1:/tmp/c_foobar
```

  Note that even if your home directory has been authorized for writes, **rsync transfers to your home directory will fail unless the "-T" or "--temp-dir" option is specified**. This is because rsync uses temporary files starting with "." during transfers, which cannot be written in your home directory. By specifying an alternate temporary directory that is authorized for writes, this problem can be avoided. For example, the following uses /tmp as the temporary directory when files are transferred to the home directory. Make sure that the temporary directory specified has enough space for the files being transferred.

```
rsync -T /tmp -e "ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q" \
    foobar cfe1:
```

- **scp** (man page)

  1. Create a file (for example, "supwrap") containing the following:

```
#!/bin/sh
 exec ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q $@
```

  2. Make the created file executable:

```
chmod 700 supwrap
```

  3. Initiate the transfer. For example:

```
scp -S ./supwrap foobar cfe1:/tmp/c_foobar
```

- **sftp** (man page)

  1. Create a file (for example, "supwrap") containing the following:

```
#!/bin/sh
    exec ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q $@
```

Note that this file is identical to the one described for scp.

2. Make the created file executable:

```
chmod 700 supwrap
```

3. Initiate the transfer. For example:

```
sftp -S ./supwrap cfe1
```

- **test** (man page)

```
ssh -Aqx -oBatchMode=yes sup.nas.nasa.gov ssh -q cfe1 test -f /tmp/c_foobar
```

SUP Troubleshooting

The following error messages may be encountered during SUP usage.

- "WARNING: Your password has expired"

  This message indicates that your current password has expired and must be changed. To change your password, you must log in to an LDAP host (e.g. Lou) through the SFEs and change your LDAP password. This change will be automatically propagated to the SUP within a few minutes.
- "Permission denied (~/.meshrc not found)"

  This message indicates that you have not created a .meshrc file in your home directory on the target host. SUP commands must be authorized for execution on each target host.
- "Permission denied (key expired)"

  SUP keys are only valid for one week from the time of generation. This message indicates that the SUP key used for authentication has expired and is no longer valid. You must generate a new SUP key or use a different SUP key before attempting another operation.
- "Permission denied (publickey,keyboard-interactive)"

  This message indicates that you have not provided the appropriate authentication credentials to the SUP. There may be several causes:

  - If you are generating a SUP key and also receive an "Error copying key..." message, you have not loaded a private key into your SSH agent corresponding to one of the public keys in the authorized_keys file uploaded to sup-key in steps 1-3 of the SUP Key Generation section. You can verify that the correct key is loaded by running "ssh-keygen -l -f uploaded_key_file" and "ssh-agent -l" and checking that the fingerprint of your uploaded key file has been loaded into your SSH agent.

- If you have specified -oBatchMode=yes on the command line, a valid SUP key may not been loaded into your SSH agent. There may also be too many keys loaded into your agent. SSH tries each key in the agent sequentially, so a valid key may still fail if it was added to the agent after a number of invalid keys greater than or equal to the login attempt limit. Check the number of keys in the agent using "ssh -l". The agent may be cleared of keys using "ssh-add -D".

- If you have specified -oPubkeyAuthentication=no, you have not provided a valid password and/or a valid securID passcode.

- "Permission denied (unauthorized command)"

  This message indicates that you have attempted an operation that is not currently authorized by the SUP. Check that the command line is valid and that the attempted command is one of the authorized commands. Certain options to authorized commands may also be disallowed, but these should never be needed in standard usage scenarios.
- Permission denied during file access (various forms)

  These messages indicate that you attempted to read or write a file for which such access is not allowed. The most common cause is forgetting to authorize directories for writes. Reads and writes of ~/.* are never permitted.

Article ID: 241
Last updated: 16 Feb, 2011
Data Storage & Transfer -> File Transfers -> Using the SUP without the SUP Client
http://www.nas.nasa.gov/hecc/support/kb/entry/241/?ajax=1